

## Research Article

# Neural Net Gains Estimation Based on an Equivalent Model

**Karen Alicia Aguilar Cruz,<sup>1</sup> José de Jesús Medel Juárez,<sup>1</sup>  
José Luis Fernández Muñoz,<sup>2</sup> and Midory Esmeralda Viguera Velázquez<sup>1</sup>**

<sup>1</sup>*Centro de Investigación en Computación, Instituto Politécnico Nacional (CIC-IPN),  
Avenida Juan de Dios Bátiz, Esq. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo,  
Delegación Gustavo A. Madero, 07738 Ciudad de México, DF, Mexico*

<sup>2</sup>*Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada U. Legaria,  
Instituto Politécnico Nacional (CICATA-Legaria-IPN), Calzada Legaria 649, Col. Irrigación,  
Delegación Miguel Hidalgo, 11500 Ciudad de México, DF, Mexico*

Correspondence should be addressed to Karen Alicia Aguilar Cruz; karen.ali320@hotmail.com

Received 30 January 2016; Revised 15 April 2016; Accepted 24 April 2016

Academic Editor: Manuel Graña

Copyright © 2016 Karen Alicia Aguilar Cruz et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A model of an Equivalent Artificial Neural Net (EANN) describes the gains set, viewed as parameters in a layer, and this consideration is a reproducible process, applicable to a neuron in a neural net (NN). The EANN helps to estimate the NN gains or parameters, so we propose two methods to determine them. The first considers a fuzzy inference combined with the traditional Kalman filter, obtaining the equivalent model and estimating in a fuzzy sense the gains matrix  $A$  and the proper gain  $K$  into the traditional filter identification. The second develops a direct estimation in state space, describing an EANN using the expected value and the recursive description of the gains estimation. Finally, a comparison of both descriptions is performed; highlighting the analytical method describes the neural net coefficients in a direct form, whereas the other technique requires selecting into the Knowledge Base (KB) the factors based on the functional error and the reference signal built with the past information of the system.

## 1. Introduction

There are different techniques for modelling a system to identify the characteristics that make it an excellent approximation. When considering the physical behaviour of the real process, these models usually lead to complex nonlinear systems, which are difficult to analyse and are simplified through relations between their input and output signals, obtaining a Black-Box (BB) description [1].

The system evolution viewed as a BB has no access to its internal properties but only to the input and output responses, without paying attention to the internal parameters that have a dynamical evolution. However, the human experiences, in many cases, give the original answer or selection (a fair process) through the intuition (if then (fuzzy logic) inferences) selecting the new parameters. These experiences provide better tolerable approximations in combination with

other theories, for example, Lyapunov, Sliding Modes, or Intelligent Systems.

Artificial Neural Nets (ANN) viewed as mathematical models, applied for complex systems [2, 3], and inspired by biological neurons operations, generate a fast answer. Nevertheless, the computer devices, compared with the human brain, have different connotations because the methods considered are faster due to robust programming instead of chemical reactions in natural conditions. In addition, the human brain has the ability of self-programming and adaptability without requiring a new programming code, adjusting its required energy levels based on natural instincts using the intuition as a great tool.

An artificial neural model (Figure 1) based on a biological neuron principle implemented in a computational model is helpful in prediction and classification of problems, pattern recognition, signal processing, estimation, and control [4]. In

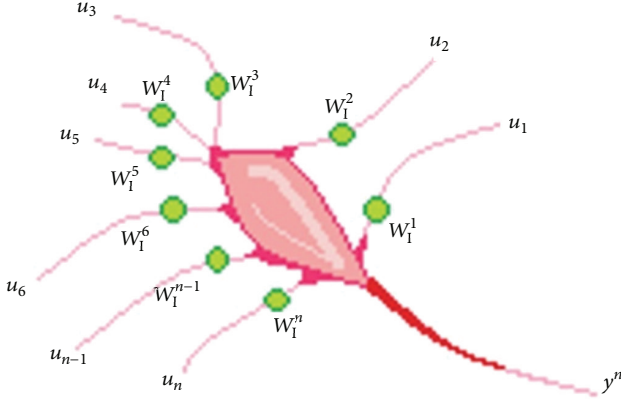


FIGURE 1: Basic neuron model.

the case when more than a neuron is connected, we obtain a MISO (Multiple Inputs and a Single Output) neural net, and similar to human adaptability, the model considered adjusts its functional parameters using learning processes for different outputs depending on the stimuli.

When analysing a system, its characteristics help us determine the best method according to our requirements, improving the convergence rate. In this paper, we compare a hybrid model and an analytical model. The first implies a fuzzy estimation combined with the Kalman filter description, and the second is optimal with systematic evaluation, considering the expected value obtained from the previous information. Both are based on an Equivalent Artificial Neural Net (EANN).

## 2. Equivalent Neural Net

In the biological sense, for neural-specific tasks, the neuron cell inputs require accomplishing adequate properties for generating a fire action in the neuron soma to obtain an accurate output in the neuron axon; in other cases, the biological system operates with the minimal energy required which allows not losing connection with other neurons.

In an ANN, we identify three principal sections: input, hidden, and output layers. The input layer is the interaction between the input signals and the first block of weights; its result becomes the input to the following coating. At the very first stage, the designer selects the weights intuitively and adjusts the following while looking for the desired response [5]. The hidden layers have a set of inputs and outputs in different stages related through the weights. The output layer

represents the convolution or binary sum of the last block of weights and its respective data. In an illustrative way, Figure 2 depicts an example of a typical ANN, with two hidden layers, input and output.

The EANN is a simplified representation of an ANN whose task is to obtain the parameters vector which allows the system to reach the desired reference signal without paying initial attention to the internal layers, focusing on the estimation procedure. It considers the total input-output signals relation trying to reduce unnecessary delays and always keeping the weight interconnection form, achieving the desired response.

As shown in Figure 2, input data is denoted by a set described as  $\{u_i : i = \overline{1, n}, n \in \mathbb{Z}_+, u \in \mathbb{R}\}$ , and the output data is denoted by  $\{y^n \in \mathbb{R}\}$ , where  $n$  represents the number of input elements. In addition, the weights or parameters are considered as  $\{W_l^k : k = \overline{1, n}, n, l \in \mathbb{Z}_+\}_{l=\overline{1, L}}$ , where  $k$  indicates the specific parameter number in a layer  $l$ . The admitted layers are within a set of functions  $\{\{f_{i,j}^l, j = \overline{1, m}, m \in \mathbb{Z}_+\}_{i=\overline{1, n}}\}_{l=\overline{1, L}}$ , interconnecting directionally from an original parameter  $i$  into layer  $l$  to a target parameter  $j$  into layer  $(l + 1)$ . Each weight from the input and output layers requires an activation function and all the hidden layers have proper activation functions connected to other weights to achieve different and specific requirements for each output stage. This description corresponds to Figure 3, where the traditional ANN connections have now simple flow diagram lines containing activation functions described as  $\{\{f_{i,j}^l, i = \overline{1, n}; n, l \in \mathbb{Z}_+\}_{l=\overline{1, L}}\}$ , where  $i$  represents the function number and  $l$  represents the layer this functions leaves.

Figure 3 presents the activation functions from the first hidden layer ( $l = 1$ ) operating with an accumulative energy  $W_1^{k=i}$  convolved with an input  $u_i$  in agreement with the following equation:

$$f_i^1 = \begin{cases} W_1^i \circ u_i & \text{if } \underline{\omega}_i \leq |u_i| \leq \overline{\omega}_i \\ 0 & \text{in other cases.} \end{cases} \quad (1)$$

The set of pairs  $\{(\underline{\omega}_i, \overline{\omega}_i) : i = \overline{1, n}, n \in \mathbb{Z}_+\}$  represents activation limit functions for  $f_i^1$ . These limits denote the minimum and maximum required energy to excite a neuron for a specific weight  $W_1^i$ , known as fire limits. In the following hidden layer (II),  $f_i^{II}$  requires that the set of inputs accomplishes the same requirements, considered for previous results; that is,

$$f_i^{II} = \begin{cases} W_{II}^i \circ (f_1^I * f_2^I * \dots * f_n^I) & \text{if } \underline{\omega}_i^{II} \leq |(f_1^I * f_2^I * \dots * f_n^I)| \leq \overline{\omega}_i^{II} \\ 0 & \text{in other cases.} \end{cases} \quad (2)$$

In (2), the binary operator “ $*$ ” represents the composition of the involved terms without indicating a particular operation.

The equivalent weights sequence allows each input to include the structure of the previous parameters in the final description. Each layer takes part in the following activation

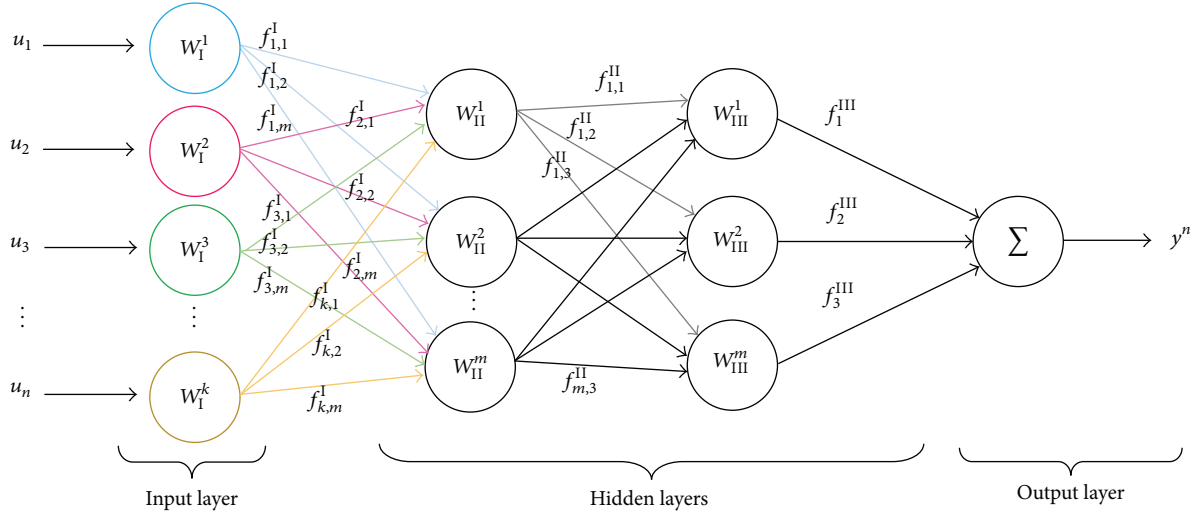


FIGURE 2: Artificial Neural Net description.

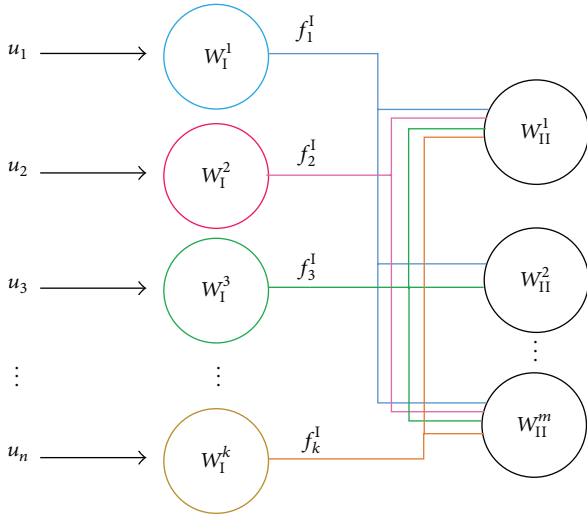


FIGURE 3: Simple description of an Artificial Neural Net through activation functions.

function due to the interaction between the new weights and the previous composed output signal. Figure 4 shows the EANN model in the simplified form.

According to [6], each neuron output has a function whose parameters are the inputs and weights for the following layer. Equation (3) expresses the influence of the previously mentioned parameters, weights, and inputs to the following layer in a recursive form, where, instead of  $y^n$ ,  $S_l^n$  describes the operation  $\sum_{i=1}^n f_i^l W_l^i u_i$  as the core neuron function:

$$S_l^n = f_n^l W_l^n u_n + M S_l^{n-1}, \quad (3)$$

where  $M$  is a proportional constant adjusting the previous layer; therefore, the model converges to the neural net development. At the final coat, we have the convolution  $y^n = ((F \circ W) \circ U)_l^n$ , which represents the neural net response.

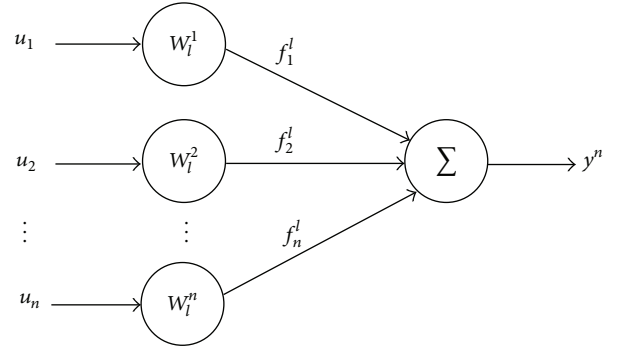


FIGURE 4: Equivalent Neural Net representation.

For computational applications, this reaction has the effect of an activation function, usually the sigmoid function.

A sophisticated ANN considers the integration of more than one EANN since its description allows using the recursive characteristics. In addition to this, the implementation of EANNs gives the possibility to restrict the number of necessary iterations to reach a reference, which is the remarkable feature in systems where time delays are considerable.

### 3. Equivalent Neural Net Using Arma Description

An ARMA (1, 1) (Autoregressive Moving Average) model is a tool used for obtaining the parameters matrix from a reference system viewed as a MISO BB; its primary structure is specified by (4) and (5), with  $n$  being the time evolution:

$$S_l^{n+1} = A S_l^n + B_l^n u_n, \quad (4)$$

$$y^n = C S_l^n, \quad (5)$$

where  $S_l^n \in R_{[0,1]}$ ,  $\{u_n\} \subseteq N(\mu_u, \sigma_u^2 < \infty)$ , and  $y^n \in R_{[0,1]}$ .

This model has observable ( $y^n$ ) and internal ( $S_l^n$ ) states, an input signal ( $u_n$ ), gains ( $B, C$ ), and internal gain ( $A$ ). The measurable state (6) in explicit form is a function of its immediate past, internal gain, and the inputs  $\{u_i\}_{i=1,n}$ . Consider

$$y^n = f(y^{n-1}, A, \{u_i\}_{i=1,n}). \quad (6)$$

In [7] the internal state using the traditional Kalman filter (KF) is described even though the internal gain  $A$  and the gain  $K^n$  are still unknown. The complexity of the filter increases because after the identification the internal gain depends on the error, which has an application in (4) for obtaining the observable signal approximation in (5), represented in the discrete form in the following equation:

$$S_l^n = S_l^{n-1} + B_l^{n-1} u_{n-1}. \quad (7)$$

By applying (7) in (4), including the still unknown internal state, we obtain (8)

$$y^n = CA S_l^{n-1} + CB_l^{n-1} u_{n-1}. \quad (8)$$

The internal state from (6) allows in (9) obtaining the internal lagged state as a measurable state function and output perturbations. Consider

$$S_l^{n-1} = C^+ y^{n-1}. \quad (9)$$

Considering (9) in (8) we determine the output in the following equation:

$$y^n = Ay^{n-1} + CB_l^{n-1} u_{n-1}. \quad (10)$$

Equation (11) represents a recursive form of (10) describing the reference system with an innovation process:

$$y^n = Ay^{n-1} + \widehat{\delta} \widehat{w}_n, \quad (11)$$

$$\widehat{\delta} \widehat{w}_n = CS_l^{n-1} u_{n-1}. \quad (12)$$

In agreement with [8], the gain ( $K^{n-1}$ ) with (12) corresponds to  $K^{n-1} \widehat{\delta} \widehat{w}_n \approx \widehat{y}^n - \widehat{A}_n y^{n-1}$ . The hybrid filter (13) considers the fuzzy parameter estimation, the gain description, and the lagged signal:

$$\widehat{y}^n = \widehat{A}_n y^{n-1} + K^{n-1} \widehat{\delta} \widehat{w}_n. \quad (13)$$

With the innovation process and the reference system, bounded by the same general Membership Function (MF) [8, 9], it is possible to estimate the explicit matrix parameters and the gain using the inference mechanisms considering the functional results and the noise properties, respectively.

#### 4. Fuzzy Gains and Estimation Properties

In the fuzzy sense, [10] presented the parameters obtained by a controller, considering a fuzzy function vector for nonlinear systems. The MIMO system found firstly the linear representation formed by a collection of MISO systems with the same inputs, reducing and simplifying its analysis.

On [11], the hybrid combination required that the identification filter adjust the parameters automatically using fuzzy logic. This adjustment needs the selection of the best values with respect to the inference, minimizing the error convergence by using heuristic techniques or the Least Square Method (LSM).

The first step in the fuzzy estimation determines the reasoning levels in accordance to the proposed MF, identified through the reference signal statistical properties. There could be triangular, sinusoidal, and impulsive or Gaussian functions, among others, to define the ranges contained in the reference signal classification.

A set of fuzzy rules (if then) forms a Fuzzy Rule Base (FRB) to interpret what requirements and process conditions are needed. Previously, it is necessary to select and introduce the best values to the Knowledge Base (KB) according to the MF, actualizing the parameters according to the reference model limited by the filter error criteria.

Using the fuzzy logic connectors into the fuzzy stage, considering the desired signal ( $y^n$ ) and the region level with respect to ( $J_n^n$ ), reducing the inference operational levels and indicators in to the MF, and selecting from the KB parameters  $\widehat{A}_n$  values actualise the hybrid filtering process. Each fuzzy filtering rule finds specific matrix parameters in each evolution [9, 12].

In the same sense, the hybrid filter considers the basic principles of a conventional Kalman digital filter using the Mean Least Square Criterion (MLSC) described as  $J_n^n = \langle e^n, e_n^T \rangle^{(1/2)}$  and, in agreement with [5], in its recursive form:

$$J_n^n = [e^n e_n^T + J_{n-1}^{n-1}]^{1/2} \in R_{[0,1)}. \quad (14)$$

According to [9], (14) presents the adequate element describing the optimal matrix parameters.

#### 5. Optimum Coefficient

For an ANN, to determine an optimal vector coefficient is necessary to consider minimizing the error and as the primary objective that the convergence tends to zero. One inconvenience is how long this optimal convergence will take to occur. A control for a recurrent NN, described in [13], was an optimum by adding an extra coefficient to compensate for the error within a small bound in an unknown necessary learning time.

Considering the fact that the last stage of a hybrid filter corresponds to the equivalent neural net from Figure 3, it is possible to determine the optimum parameters for the neural weights obtaining the best output approximation to the reference signal by an analytical process.

Based on BB concepts, the input signals  $\{u_i\}_{i=1,N}$  represented by the matrix  $x^n_{[N \times 1]}$  and the output of the system  $y^n$  are the known parameters. In this sense, we need a synthesis process to calculate the matrix values  $A_{[1 \times N]}$  representing the weights in the neural layer.

Having  $y^n = Ax^n$  as an ARMA model and the process considering stochastic properties, we use the mathematical expectation in the probabilistic sense obtaining information

about the process. So  $\widehat{A}_n := E\{y^n x_n^T\} [E\{x_n x_n^T\}]^+$ , where the symbols  $T$  and  $+$  represent the transpose and pseudoinverse operators, respectively.

If  $y^n$  is the reference signal which helps us get the parameters, then we apply these values to find the output  $y^{\text{iden},n}$ , and their comparison gives the identification error  $e^n := y^n - y^{\text{iden},n}$  and its functional error  $J_n^n := \langle e^n, e_n^T \rangle$  tending to zero due to the values being considered optimums.

To demonstrate this, from Figure 4, the output is observed as

$$y^n = w_1^1 u_{1_n} + w_1^2 u_{2_n} + \cdots + w_1^N u_{N_n}. \quad (15)$$

In addition, seeing  $y^n$  as the reference or target signal defined as

$$A := [w_1^1 \ w_1^2 \ \cdots \ w_1^N]_{[1 \times N]}; \quad (16)$$

$$x_n^T := [u_{1_n} \ u_{2_n} \ \cdots \ u_{N_n}]_{[1 \times N]},$$

we have the following form:

$$y^n = Ax^n. \quad (17)$$

Considering  $x_n^T$  is a stochastic input formed in distribution sense by  $\{(u_{i_n})^T \subseteq N(\mu, \sigma^2 < \infty)\}_{i=1, \dots, N}$ , the parameters are represented by  $A$  and the output signal is represented by  $y^n$ ; the BB system scheme allows estimating the parameters set through its time evolution in a probabilistic sense. Consider

$$E\{y^n_{[1]} x_{n[1 \times N]}^T\} = E\{A_{[1 \times N]} x^n_{[N \times 1]} x_{n[1 \times N]}^T\}. \quad (18)$$

Due to the weights being constants for an instant of execution time  $n$  and considering the mathematical expectation properties, it is possible to obtain the matrix estimation known as  $\widehat{A}_n$ , indicating that this new array value is the matrix estimation. Consider

$$\widehat{A}_{n[1 \times N]} \cong E\{y^n_{[1]} x_{n[1 \times N]}^T\} (E\{x^n_{[N \times 1]} x_{n[1 \times N]}^T\})^+. \quad (19)$$

For a discrete system (19) with infinite enumerable elements, the mathematical expectation has the following form:

$$\widehat{A}_{n[1 \times N]} \cong \left( \frac{1}{n} \sum_{i=1}^n y^i_{[1]} x_{i[1 \times N]}^T \right)_{[1 \times N]} \cdot \left( \frac{1}{n} \sum_{i=1}^n x^i_{[N \times 1]} x_{i[1 \times N]}^T \right)^+_{[N \times N]}. \quad (20)$$

By replacing  $\widehat{A}_{n[1 \times N]}$  in (17), we obtain a new output state of  $y^n$  which we call identification symbolically described as  $y^{\text{iden},n}$ ; it represents the output including the effects of the estimated weights values.

The difference between the identified signal and the reference signal gives the following identified error:

$$e^n = y^n - y^{\text{iden},n}. \quad (21)$$

In order to express (20) recursively, the first and second terms are replaced with  $P^n$  and  $Q_n$ , respectively, defined as follows:

$$P^n = \left( \frac{1}{n} \right) \sum_{i=1}^n y^i x_i^T, \quad (22)$$

$$Q_n = \left( \frac{1}{n} \right) \sum_{i=1}^n x^i x_i^T. \quad (23)$$

Considering (22) and (23) in (20), (24) and its delayed form (25) for stable conditions are obtained:

$$\widehat{A}_n = P^n Q_n^+, \quad (24)$$

$$\widehat{A}_{n-1} = P^{n-1} Q_{n-1}^+. \quad (25)$$

Developing (22) in recursive manner has the following equation:

$$P^n = \left( \frac{1}{n} \right) \left( y^n x_n^T + \sum_{i=1}^{n-1} y^i x_i^T \right). \quad (26)$$

Considering stationary conditions for (22) delayed has

$$P^{n-1} = \left( \frac{1}{n-1} \right) \sum_{i=1}^{n-1} y^i x_i^T. \quad (27)$$

Rewriting (26) in terms of (27), we have (28) and its block diagram representation shown in Figure 5:

$$P^n = \left( \frac{1}{n} \right) (y^n x_n^T + (n-1) P^{n-1}). \quad (28)$$

Expanding (28) and ordering with respect to  $P^{n-1}$ , we have the following equation:

$$P^n = \left( \frac{n-1}{n} \right) P^{n-1} + \left( \frac{1}{n} \right) y^n x_n^T. \quad (29)$$

Now, applying (29) in (24), we have the following estimation:

$$\widehat{A}_n = \left( \frac{n-1}{n} \right) P^{n-1} Q_n^+ + \left( \frac{1}{n} \right) y^n x_n^T Q_n^+. \quad (30)$$

Remembering that (25) in stationary conditions is the estimation delayed and applying it in (30) yields the following:

$$\widehat{A}_n = \left( \frac{n-1}{n} \right) \widehat{A}_{n-1} (Q_{n-1} Q_n^+) + \left( \frac{1}{n} \right) y^n x_n^T Q_n^+. \quad (31)$$

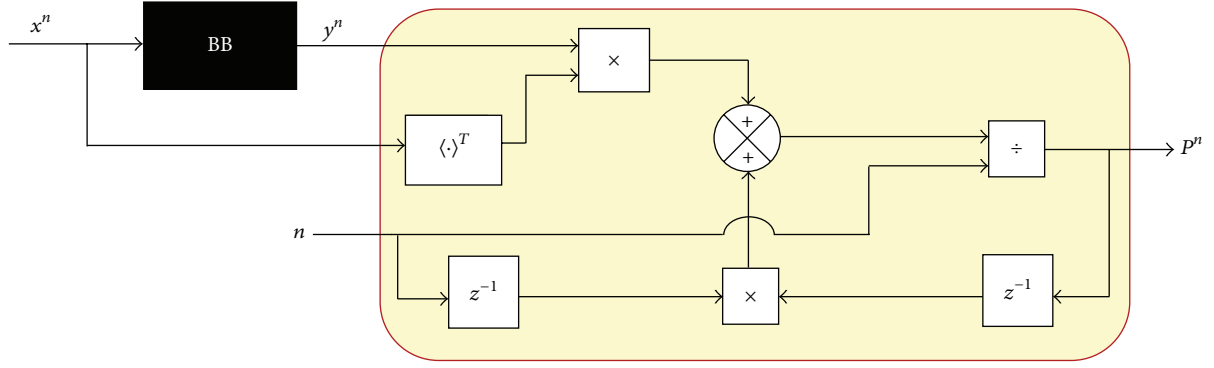
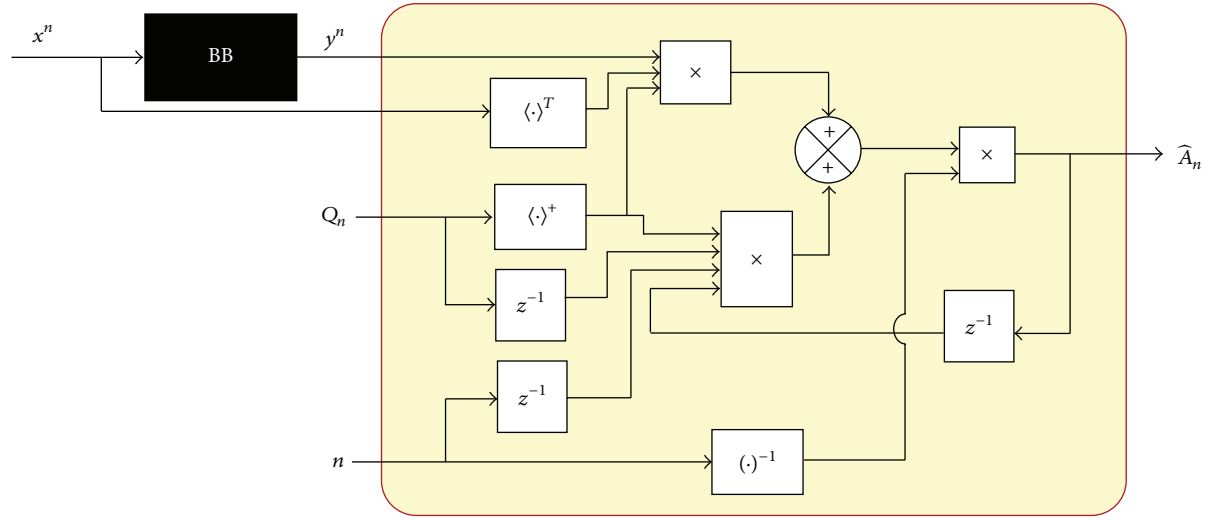
Using (31) in (24), we obtain the parameter vector in recursive form (32). The block diagram representing  $\widehat{A}_n$  parameter using (31) is in Figure 6. Consider

$$\widehat{A}_n = \widehat{A}_{n-1} \alpha^{n-1} + \beta_{n-1}, \quad (32)$$

where  $\alpha^{n-1} = ((n-1)/n) Q_{n-1} Q_n^+$  and  $\beta_{n-1} = (1/n) y^n x_n^T Q_n^+$ .

As (31) includes (23) in its description, it is necessary to build its recursive form similar to the obtainment of (28); then



FIGURE 5: Obtainment of  $P^n$ , recursive.FIGURE 6: Parameter  $\hat{A}_n$  block diagram.

we have (33) and its block diagram representation shown in Figure 7. Consider

$$Q_n = \left( \frac{1}{n} \right) [x^n x_n^T + (n-1) Q_{n-1}]. \quad (33)$$

Finally, replacing (32) in (17), the identified output is the following equation:

$$y^{\text{iden},n} = \hat{A}_n x^n. \quad (34)$$

Figure 8 represents the interaction between the inputs and the resulting error, which has better convergence due to the null error, determined for an instant by the best parameters values.

## 6. Hybrid Mechanism of Inference

Figure 9 provides the block diagram of a hybrid filter that combines fuzzy inferences with the EANN ARMA model description, instead of the logical block from Figure 8, to determine the adequate matrix parameters. The reference model considered is a BB giving the reference signal  $y^n$ . The distribution curve of this signal denotes the intervals where

the MF must be; then, the degree of membership obtained by Mamdani with fuzzy inferences has access to the Knowledge Base (KB), determining the parameters of the model, making the convergence, and minimizing the error in a distribution sense.

## 7. Results

The performed simulation considers a comparison between both methods giving a better idea of how they approximate to the reference signal. The reference model output  $y^n$  considered nonstationary conditions, noise sequences bounded by a distribution function, and, on average, constant mean expected value and variance. The variations in the signal have a periodic signal with smooth random perturbations.

The first part of the simulation considered the hybrid filter, applying inferences obtaining (13) as the signal output identification. Figure 10 shows the fuzzy inference process, where it is possible to identify the functional error given by (14), useful to estimate the coefficients for the ANN. The distribution curves defined the MFs having different operational levels represented through three and seven MFs, corresponding to  $y^n$  and  $J_n^n$ , respectively.

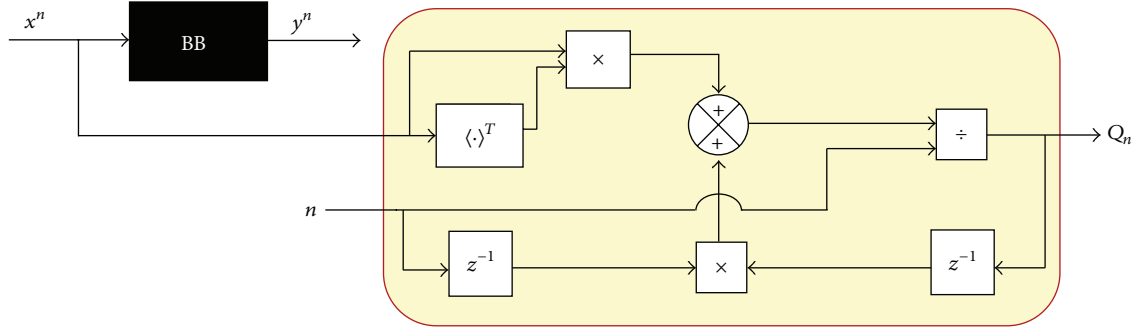
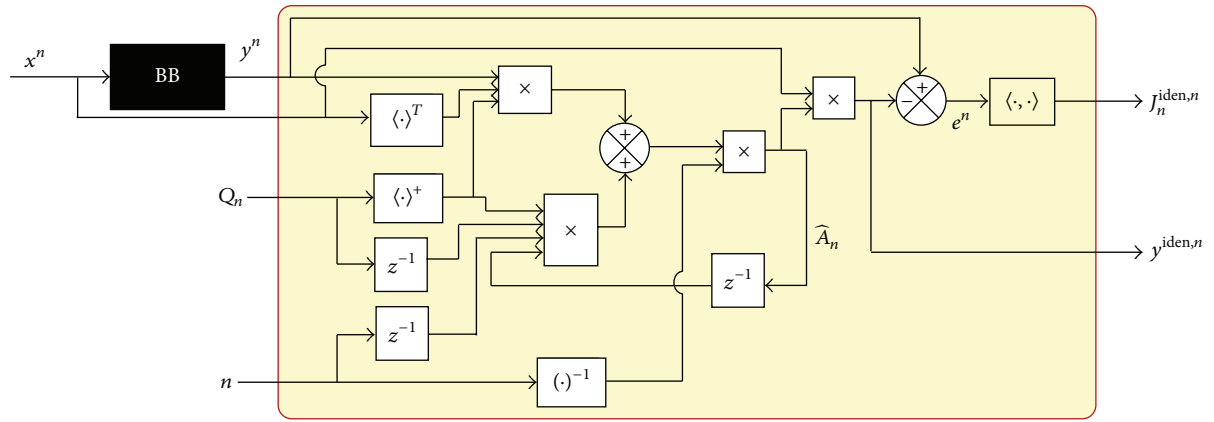
FIGURE 7: Obtainment of  $Q_n$ , recursive.

FIGURE 8: Block diagram of the analytical process.

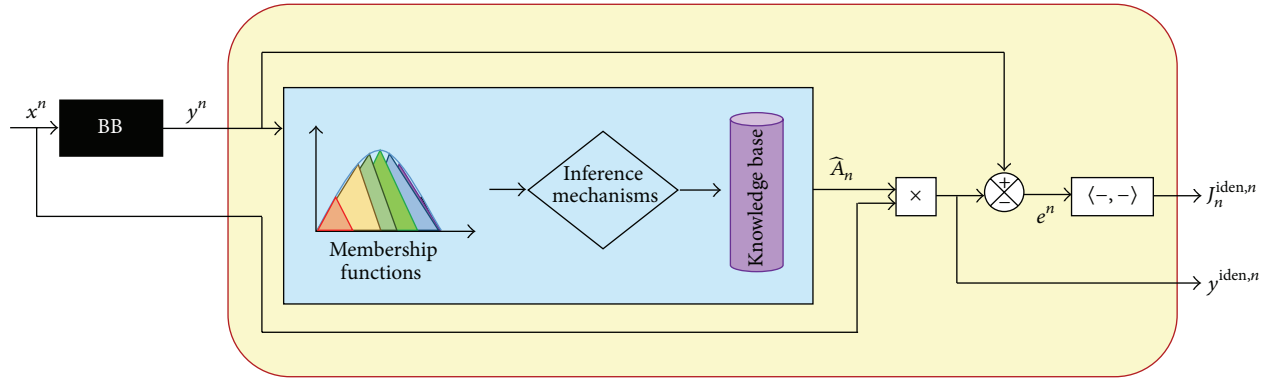


FIGURE 9: Block diagram of a hybrid filter with Fuzzy inference for an EANN model through ARMA description.

These MFs are results of associated proper inference mechanisms to select parameters  $\hat{A}_n$  and gain  $K^n$  through the MFs and the KBs, affecting the final identified output  $\hat{y}^n$ . As an example, Figure 11 presents a three-dimensional KB integrated by sets: gain  $\{K^n\}$ , reference signal  $\{y^n\}$ , and functional error  $\{j_n^n\}$ . This KB helps us determine gain  $K^n$  through the reference and operational error considering our expertise. The KB for  $\hat{A}_n$  has a similar structure.

The analytical method uses the block diagram presented in Figure 8 having a delay in execution time due to the time

state operations but within fewer process stages due to the fact that it does not require feedback from the functional error.

Our objective was to determine the internal parameters; Figure 12 compares the reference signal parameters to those estimated with both methods. The polar representation allows observing the components of the parameters; where it is possible to see, none of them leaves the unit circle.

When applying the estimation into the hybrid system, the response is as shown in Figure 13, which presents the response following the tendency of the reference.

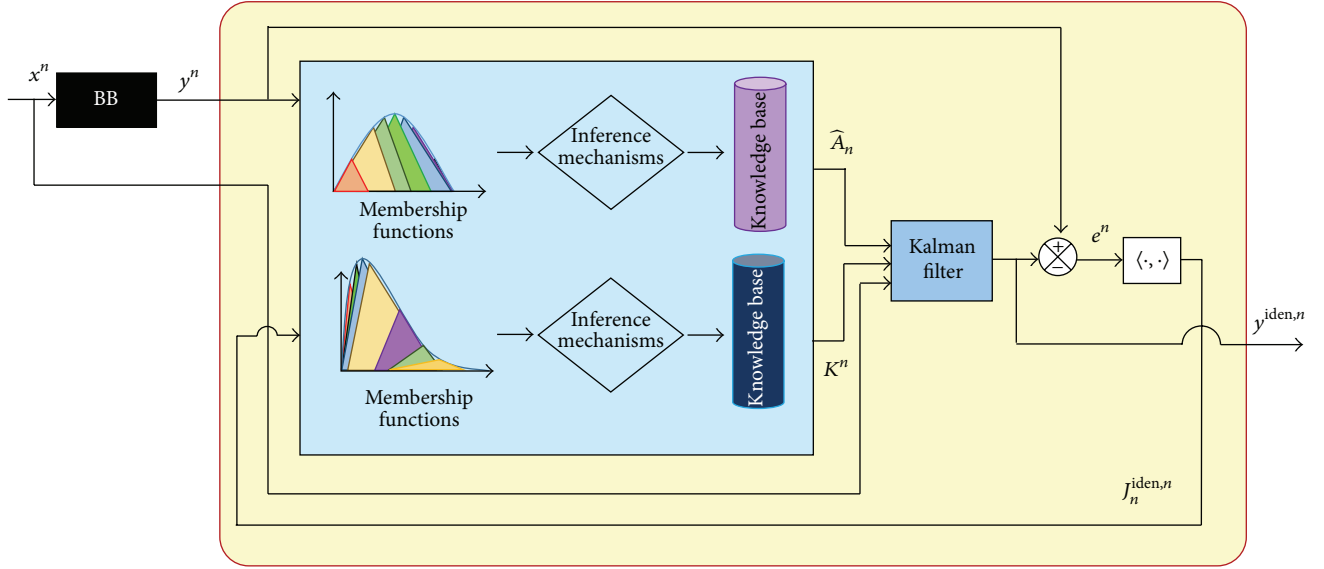


FIGURE 10: Inference process in the hybrid filter.

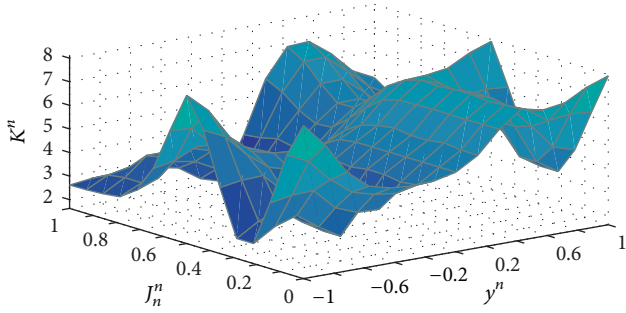
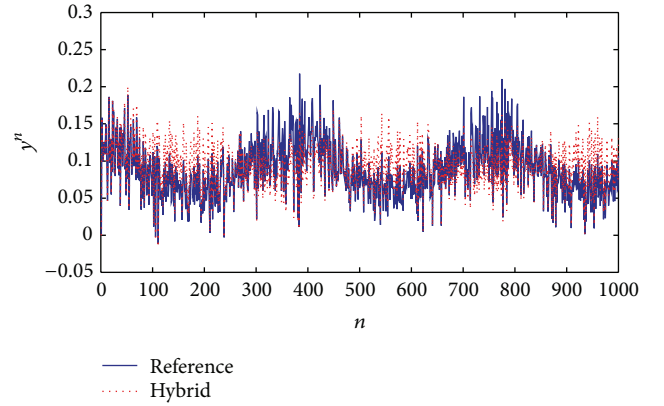
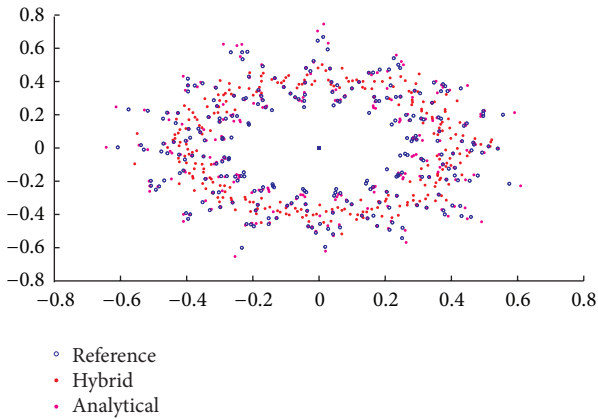
FIGURE 11: Example of a three-dimensional Knowledge Base to obtain the gain ( $K^n$ ) through the reference signal ( $y^n$ ) and the functional error ( $J_n^n$ ).

FIGURE 13: Comparison of the system response: reference signal (blue) and its hybrid approximation (red).

FIGURE 12: Comparison of the internal system parameters in a polar graph: reference signal parameters  $A_n$  (blue), estimated signal parameters ( $\hat{A}_n$ ) through hybrid estimation (red), and analytical estimation (magenta).

The analytical method provides the response in Figure 14. The previous graphics, Figures 12–14, were obtained considering a reference system with variable parameters and random noise. In order to better identify how the approximations converge to the reference, we have Figure 15 which presents a graphic segment showing more clearly both approximations to a reference system response, with also variable parameters but without random noise.

From Figure 15, Figure 16 compares the convergence considering the functional error (14) from both methods. In this case, the reference is near to zero as a constant value due to the estimations considered as optimum.

## 8. Conclusion

An Equivalent Artificial Neural Net (EANN) was considered describing its parameter through a Black Box (BB) analysis



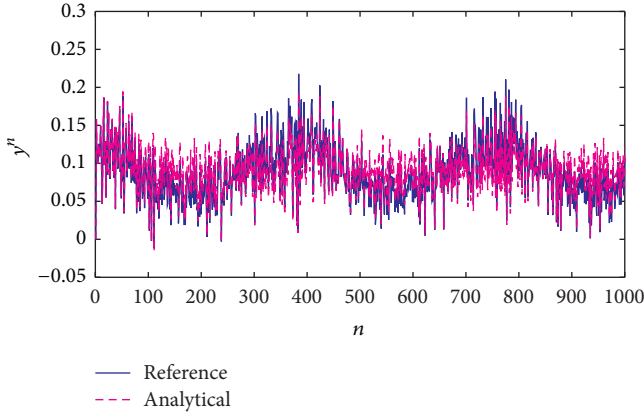


FIGURE 14: Comparison of the system response: reference signal (blue) and its analytical approximation (magenta).

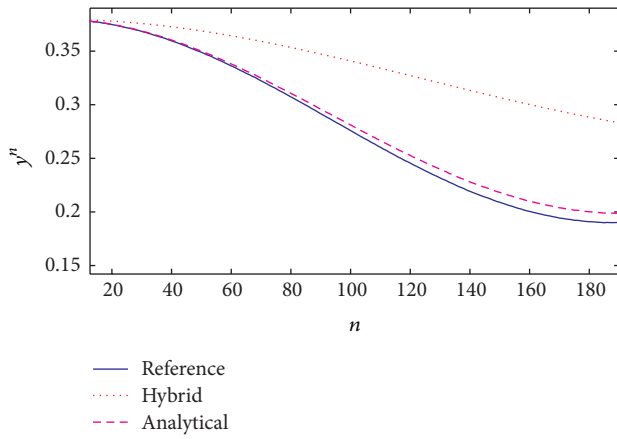


FIGURE 15: Comparison of the system response convergence for a system with variable parameters and no random noise.

using two different approximations, hybrid and analytical techniques.

For the fuzzy estimation, the best option was to consider the error properties and, in this method, the response signal was adjusted according to the reference. The fuzzy evaluation allowed the description of the coefficients and gain which affect the Kalman filter, improving the identification process according to the Multiple Inputs and Single Output (MISO) model changes with perturbations. The parameter and gain selection, using an intelligent system with classification levels, allowed selecting into the KBs the best coefficients that positively affected the filter evolution. This method does not have an exact approximation, but it is good enough on average, as shown in Figure 12, and in distribution (Figures 13 and 15) if we consider that its response converged to a particular region different from zero.

The second method used the analytical approximation, converging at almost all points to the system parameters and the reference (Figures 12 and 14) so that the expected results were a minimum functional error through time. We considered that the null error corresponded to the low energy limit, which is not zero in the neurons to avoid the total loss

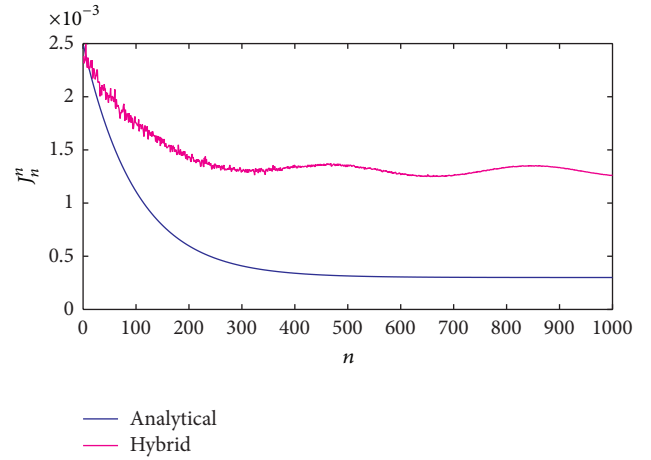


FIGURE 16: Convergence of the functional error. Comparison between the hybrid (magenta) and analytical (blue) evaluations.

of connection. This method had a better approximation to the reference but achieved the minimum error only in the numerable infinite. Even though this estimation does not consider the error feedback as the first method does, its response continues being adequate when external perturbations affect the system.

A sophisticated ANN could be represented by the integration of more than one EANN due to the fact that its description allows it to consider more than one layer because it has recursive characteristics. In addition to this, the implementation of EANNs gives the possibility of having more control on the number of necessary iterations to reach a reference; this is relevant to systems where restrictions in time delays are considered essential.

Globally, both methods presented good approximations, as shown in Figure 16, with unique characteristics identifying differences between the hybrid and analytical methods.

## Competing Interests

The authors declare that there are no competing interests regarding the publication of this paper.

## Acknowledgments

The authors wish to thank the Instituto Politécnico Nacional (IPN) and the Consejo Nacional de Ciencia y Tecnología (CONACYT) for their support while carrying out their research work. The Instituto Politécnico Nacional supported the paper through Project no. SIP20160382.

## References

- [1] H. M. Romero Ugalde, J.-C. Carmona, J. Reyes-Reyes, V. M. Alvarado, and J. Mantilla, "Computational cost improvement of neural network models in black box nonlinear system identification," *Neurocomputing*, vol. 166, pp. 96–108, 2015.
- [2] M. Witters and J. Swevers, "Black-box model identification for a continuously variable, electro-hydraulic semi-active damper,"

- Mechanical Systems and Signal Processing*, vol. 24, no. 1, pp. 4–18, 2010.
- [3] B. Subudhi and D. Jena, “A differential evolution based neural network approach to nonlinear system identification,” *Applied Soft Computing*, vol. 11, no. 1, pp. 861–871, 2011.
  - [4] K. M. Passino and S. Yurkovich, *Fuzzy Control*, Addison-Wesley, New York, NY, USA, 1998.
  - [5] J. C. G. Infante, J. J. M. Juárez, and P. G. López, “Filtrado digital difuso en tiempo real,” *Computación y Sistemas*, vol. 11, no. 4, pp. 390–401, 2008.
  - [6] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, “Real-time learning capability of neural networks,” *IEEE Transactions on Neural Networks*, vol. 17, no. 4, pp. 863–878, 2006.
  - [7] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.
  - [8] T. Takagi and M. Sugeno, “Fuzzy identification of systems and its applications to modeling and control,” *IEEE Transactions on Systems, Man and Cybernetics*, vol. 15, no. 1, pp. 116–132, 1985.
  - [9] J. J. Medel Juárez, J. C. García Infante, and P. Guevara López, “Real-Time Fuzzy Digital Filters (RTFDF) properties for siso systems,” *Automatic Control and Computer Sciences*, vol. 42, no. 1, pp. 26–34, 2008.
  - [10] Z. Huaguang, L. Cai, and Z. Bien, “A fuzzy basis function vector-based multivariable adaptive controller for nonlinear systems,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 30, no. 1, pp. 210–217, 2000.
  - [11] J. C. Garcia, J. J. Medel, and J. C. Sanchez, “Neural fuzzy digital filtering: multivariate identifier filters involving multiple inputs and multiple outputs (MIMO),” *Ingenieria e Investigacion*, vol. 31, no. 1, pp. 184–192, 2011.
  - [12] R. M. Dudley, *Real Analysis and Probability*, Cambridge University Press, Cambridge, UK, 2004.
  - [13] H. Zhang, L. Cui, X. Zhang, and Y. Luo, “Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method,” *IEEE Transactions on Neural Networks*, vol. 22, no. 12, pp. 2226–2236, 2011.